Combining Language Models with NLP and Interactive Query Expansion

Eric SanJuan[1] and Fidelia Ibekwe-SanJuan[2]

[1] LIA & IUT STID, Université d'Avignon
339, chemin des Meinajaries, Agroparc BP 1228,
84911 Avignon Cedex 9, France.
eric.sanjuan@univ-avignon.fr
[2] ELICO, Université de Lyon 3
4, Cours Albert Thomas, 69008 Lyon, France.
fidelia.ibekwe-sanjuan@univ-lyon3.fr

Abstract. Following our previous participation in INEX 2008 Ad-hoc track, we continue to address both standard and focused retrieval tasks based on comprehensible language models and interactive query expansion (IQE). Query topics are expanded using an initial set of Multiword Terms (MWTs) selected from top n ranked documents. In this experiment, we extract MWTs from article titles, narrative field and automatically generated summaries. We combined the initial set of MWTs obtained in an IQE process with automatic query expansion (AQE) using language models and smoothing mechanism. We chose as baseline the Indri IR engine based on the language model using Dirichlet smoothing. We also compare the performance of bag of word approaches (TFIDF and BM25) to search strategies elaborated using language model and query expansion (QE). The experiment is carried out on all INEX 2009 Ad-hoc tasks.

1 Introduction

This year (2009) represents our second participation in the INEX Ad-hoc track. The three tasks defined in the previous years were maintained: focused retrieval (element, passage), Relevant-in-Context (RiC), Best-in-Context (BiC). A fourth task called "thorough task" was added to this year's edition. The thorough task can be viewed as the generic form of the focused task in that systems are allowed to retrieve overlapping elements whereas this is not allowed in the focused task. In the 2008 edition, we explored the effectiveness of NLP, in particular that of multiword terms (MWTs) combined with query expansion mechanisms - Automatic Query Expansion (AQE) and Interactive Query Expansion (IQE). Previous experiments in IR have sought to determine the effectiveness of NLP in IR. A study by [1] concluded that the issue of whether NLP and longer phrases would improve retrieval effectiveness depended more on query representation rather than on document representation within IR models because no matter how rich and elaborate the document representation, a poor representation of

the information need (short queries of 1-2 words) will ultimately lead to poor retrieval performance.

A few years later, [2] applied NLP in order to extract noun phrases (NPs) used in an IQE process. The IQE approach described in her study shares similar points with that of [1] except that instead of using the abstracts of the top n-ranked documents to expand the queries, [2] extracted NPs from query topics using a part-of-speech tagger and a chunker. She tested different term weighting functions for selecting the NPs: idf, C-value and log-likelihood. We refer the reader to [3] for a detailed description and comparison of these measures. The ranked lists of NPs were displayed to the users who selected the ones that best described the information need expressed in the topics. Documents were then ranked based on the expanded query and on the BM25 probabilistic model [4]. By setting optimal parameters, the IQE experiment in [2] showed significant precision gains but surprisingly only from high recall levels.

Based on these earlier findings and on our own performance in 2008's Ad-Hoc track evaluation, we pursue our investigation of the effectiveness of representing queries with MultiWord Terms (MWTs). MWTs is understood here in the sense defined in computational terminology [5] as textual denominations of concepts and objects in a specialized field. Terms are linguistic units (words or phrases) which taken out of context, refer to existing concepts or objects of a given field. As such, they come from a specialized terminology or vocabulary [6]. MWTs, alongside noun phrases, have the potential of disambiguating the meaning of the query terms out of context better than single words or statistically-derived n-grams and text spans. In this sense, MWTs cannot be reduced to words or word sequences that are not linguistically and terminologically grounded. Our approach was successfully tested on two corpora: the TREC Enterprise track 2007 and 2008 collections, and INEX 2008 Ad-hoc track [7] but only at the document level.

We ran search strategies implementing IQE based on terms from different fields of the topic (title, phrase, narrative). We tested many new features in the 2009 edition including:

- XML element retrieval. In 2008 edition, we only did full article retrieval;
- more advanced NLP approaches including automatic multi-document summarization as additional source of expansion terms;
- expansion of terms based on related title documents from Wikipedia;
- comparison of other IR models, namely bag of word models (TFIDF, BM25) without query expansion (QE);
- a combination of different query expansion mechanisms (IQE+AQE, IQE alone, AQE alone) with the language model implemented in Indri.

Our query expansion process runs as follows. First a seed query consisting of the title field is sent to the Indri search engine which served as our baseline. The system returns a ranked list of documents. Our system automatically extracts MWTs from the top n-ranked documents and from the topic fields (title, phrase, narrative). The expanded query resulting from the IQE process is further expanded using the automatic query expansion process (AQE) implemented in

Indri. Indri is based on standard IR Language Models for document ranking. Our system also generates automatic summaries from these top ranked documents and resulting MWTs. Thus the user has multiple sources - topic fields or top ranked documents, from which to select MWTs with which to expand the initial seed query in an Interactive Query Expansion (IQE) process. Our aim was to set up a comprehensive experimental framework in which competing models and techniques could be compared. Our IR system thus offers a rich framework in which language models are compared against bag of word models in combination with different query expansion techniques (IQE, AQE) as well as advanced NLP techniques (MWT extraction, automatic document summarization). A novelty in INEX 2009's Ad-Hoc track is that a new phrase (ph) field has been added to the topic description fields. These phrases are quite similar to the MWTs we automatically extract from the other topic fields. This may have an impact on the performance of some of our IR strategies. We will discuss this issue further in the §4.

The rest of the paper is structured as follows: section §2 presents the language model and its application to the IR tasks; section §3 presents the results on the Wikipedia collection in the INEX 2009 Ad-hoc track; finally, section §4 discusses the lessons learned from these experiments.

2 Probabilistic IR Model

2.1 Language Model

Language models are widely used in NLP and IR applications [8, 4]. In the case of IR, smoothing methods play a fundamental role [9]. We first describe the probability model that we use.

Document Representation: probabilistic space and smoothing Let us consider a finite collection $\mathcal{D}$ of documents, each document D being considered as a sequence $(D_1, ..., D_{|D|})$ of $|D|$ terms D_i from a language $\mathcal{L}$, i.e. $\mathcal{D}$ is an element of $\mathcal{L}^\star$, the set of all finite sequences of elements in $\mathcal{L}$. Our formal framework is the following probabilistic space $(\Omega, \wp(\Omega), P)$ where Ω is the set of all occurrences of terms from $\mathcal{L}$ in some document $D \in \mathcal{D}$ and P is the uniform distribution over Ω. Language Models (LMs) for IR rely on the estimation of the a priori probability $P_D(q)$ of finding a term $q \in \mathcal{L}$ in a document $D \in \mathcal{D}$. We chose the Dirichlet smoothing method because it can be viewed as a maximum *a priori* document probability distribution. Given an integer μ, it is defined as:

$$P_D(q) = \frac{f_{q,D} + \mu \times P(q)}{|D| + \mu} \tag{1}$$

In the present experiment, documents can be full wikipedia articles, sections or paragraphs. Each of them define a different probabilistic space that we combine in our runs.

Query Representation and ranking functions Our purpose is to test the efficiency of MWTs in standard and focused retrieval compared to a bag-of-word model or statistically-derived phrases. For that, we consider phrases (instead of single terms) and a simple way of combining them. Given a phrase $s = (s_0, ..., s_n)$ and an integer k, we formally define the probability of finding the sequence s in the corpus with at most k insertions of terms in the following way. For any document D and integer k, we denote by $[s]_{D,k}$ the subset of $D_i \in D$ such that: $D_i = s_1$ and there exists n integers $i < x_1, ..., x_n \leq i + n + k$ such that for each $1 \leq j \leq n$ we have $s_j = D_{x_j}$.

We can now easily extend the definition of probabilities P and P_D to phrases s by setting $P(s) = P([s]_{.,k})$ and $P_D(s) = P_D([s]_{D,k})$. Now, to consider queries that are set of phrases, we simply combine them using a weighted geometric mean as in [10] for some sequence $w = (w_1, ..., w_n)$ of positive reals. Unless stated otherwise, we suppose that $w = (1, ..., 1)$, i.e. the normal geometric mean. Therefore, given a sequence of weighted phrases $Q = \{(s_1, w_1), ..., (s_n, w_n)\}$ as query, we rank documents according to the following scoring function $\Delta_Q(D)$ defined by:

$$\Delta_Q(D) = \prod_{i=1}^{n} (P_D(s_i))^{\frac{w_i}{\sum_{j=1}^{n} w_j}} \tag{2}$$

$$\stackrel{\text{rank}}{=} \sum_{i=1}^{n} \left(\frac{w_i}{\sum_{j=1}^{n} w_j} \times \log(P_D(s_i)) \right) \tag{3}$$

This plain document ranking can easily be computed using any passage information retrieval engine. We chose for this purpose the Indri engine [11] since it combines a language model (LM) [8] with a bayesian network approach which can handle complex queries [10]. However, in our experiments, we use only a very small subset of the weighting and ranking functionalities available in Indri.

2.2 Query Expansion

We propose a simple QE process starting with an approximate short query $Q_{T,\mathcal{S}}$ of the form $(T, \mathcal{S})$ where $T = (t_1, ..., t_k)$ is an approximate document title consisting of a sequence of k words, followed by a possibly empty set of phrases: $\mathcal{S} = \{S_1, ..., S_i\}$ for some $i \geq 0$. In our case, each S_i will be a MWT.

Baseline document ranking function By default, we rank documents according to :

$$\Delta_{T,\mathcal{S}} = \Delta_T \times \prod_{i=1}^{|\mathcal{S}|} \Delta_{S_i} \tag{4}$$

Therefore, the larger $\mathcal{S}$ is, the less the title part T is taken into account. Indeed, $\mathcal{S}$ consists of a coherent set of MWTs found in a phrase query field

or chosen by the user. If the query can be expanded by coherent clusters of terms, then we are no more in the situation of a vague information need and documents should be ranked according to precise MWTs. For our baseline, we generally consider $\mathcal{S}$ to be made of the phrases given in the query.

Interactive Query Expansion Process The IQE process is implemented on a html interface available at *http://master.termwatch.es/*. Given an INEX topic identifier, this interface uses the title field as the seed query. The interface is divided into two sections:

topic section: a column displays terms automatically extracted from the topic description fields (title, phrase, narrative). A second column in this section displays titles of documents related to the query. These are titles of Wikipedia documents found to be related to the seed query terms by the language model implemented in Indri. The user can then select terms either from topic fields (title, phrase, narrative) and/or from related titles.

document summary section: this section displays short summaries from the top twenty ranked documents of Δ_Q ranking together with the document title and the MWTs extracted from the summary. The summaries are automatically generated using a variant of TextRank algorithm. The user can select MWTs in context (inside summaries) or directly from a list without looking at the sentence from which they were extracted.

MWTs are extracted from the summaries based on shallow parsing and proposed as possible query expansions. The user selects all or a subset $\mathcal{S}'$ of them. This leads to acquiring sets of synonyms, abbreviations, hypernyms, hyponyms and associated terms with which to expand the original query terms. The selected multiword terms S'_i are added to the initial set $\mathcal{S}$ to form a new query $Q' = Q_{T,\mathcal{S}\cup\mathcal{S}'}$ leading to a new ranking $\Delta_{Q'}$ computed as in §2.2.

Automatic Query Expansion we also consider Automatic Query Expansion (AQE) to be used with or without IQE. In our model, it consists in the following: let $D_1, ..., D_K$ be the top ranked documents by the initial query Q. Let $C = \cup_{i=1}^{K} D_i$ be the concatenation of these K top ranked documents. Terms c occurring in D can be ranked according to $P_C(c)$ as defined by equation (1). We consider the set E of the N terms $\{c_1, ..., c_N\}$ with the highest probability $P_C(c_i)$. We then consider the new ranking function Δ'_Q defined by $\Delta'_Q = \Delta_Q^{\lambda} \times \Delta_E^{1-\lambda}$ where $\lambda \in [0, 1]$.

Unless stated otherwise, we take $K = 4$, $N = 50$ and $\lambda = 0.1$ since these were the parameters that gave the best results on previous INEX 2008 ad-hoc track.

We now explore in which context IQE based on MWTs is effective. Our baseline is an automatic document retrieval based on equation 2 in §2.1.

3 Results

We submitted eight runs to the official INEX evaluation: one automatic and one manual for each of the four tasks (focused, thorough, BiC, RiC). Our Relevant-in-Context (RiC) runs were disqualified because they had overlapping elements. Here we focus on analysing results from focused and thorough tasks. Focused task is measured based on interpolated precision at 1% of recall (iP[0.01]) while thorough is measured based on Mean Average interpolated Precision (MAiP), so the two are complementary. Moreover, on document retrieval, computing MAiP on focused results or on thorough's is equivalent.

We compare our officially submitted runs to additional ones we generated after the official evaluation. Among them, are two baselines runs. It appears that these baseline runs outperform our submitted runs based on the qrels released by the organizers. Our runs combine features from the following:

Xml : these runs retrieve XML elements, not full articles. Each element is evaluated in the probabilistic space of all elements sharing the same tag. Elements are then ranked by decreasing probability. The following elements were considered: b, bdy, category, causal_agent, country, entry, group, image, it, list, location, p, person, physical_entity, sec, software, table, title.

Doc : only full articles are retrieved.

AQE : Automatic Query Expansion is performed.

ph : the query is expanded based on the phrases furnished this year in the topic fields. These phrases are similar to MWTs.

IQE : Interactive Query Expansion (IQE) is performed based on the interface described previously (see §2.2).

Ti : elements in documents whose title overlaps the initial query or its expansion terms are favoured.

All our submitted runs were using a default language model. After the official evaluation, we generated runs based on other ranking methods, namely TFIDF and BM25 that are also implemented in the Indri system. We also test the impact of stemming on these different methods.

3.1 Search strategies

We consider the following runs. They are all based on LM and they all use the topic Title and PhraseTitle fields.

Lyon3LIAautolmnt combines **Xml** element extraction, **Ti** heuristic, and **AQE**. It was submitted to the thorough task.

Lyon3LIAmanlmnt adds IQE on the top of the previous one and was also submitted to the thorough task.

Lyon3LIAautoQE is similar to Lyon3LIAmanlmnt but retrieves documents **Doc** instead of XML elements. It was submitted to the focused task.

Lyon3LIAmanQE adds IQE on the top of the previous one and was also submitted to the focused task.

LMDoc baseline run was not submitted. It retrieves full documents without using any of **Ti**, **AQE**, nor **IQE**.

LMDocIQE the same baseline run with IQE, also not submitted.

Table 1 summarizes these runs and gives their IP[0.01] and MAiP scores.

Name of the run	XML	Doc	ph	Ti	AQE	IQE	Submitted	IP[0.01]	MAiP
Lyon3LIAmanlmnt	×	-	×	×	×	×	thorough	0.4956	0.2496
Lyon3LIAmanQE	-	×	×	×	×	×	thorough	0.4861	0.2522
Lyon3LIAautolmnt	×	-	×	×	×	-	focus	0.4646	0.2329
Lyon3LIAautoQE	-	×	×	×	×	-	focus	0.4645	0.2400
LMDocIQE	-	×	×	-	-	×	-	0.5840	0.2946
LMDoc	-	×	×	-	-	-	-	0.5527	0.2826

Table 1. Results of submitted runs and two non submitted baselines. All of them use the Language Model.

It appears that Ti and AQE degraded the results since the non submitted baselines LMDocIQE and LMDoc outperformed all submitted runs. Retrieving Xml elements lightly improves iP[0.01] score but degrades MAiP. However these differences are not statistically significant (paired t-test, p-value=0.1). Following our observation in 2008's edition, adding IQE improved scores. However, on this year's corpus, the improvement is not significant. None of these differences are statistically significant but we can check on precision/recall curves if there is a general tendency. Figure 1 shows the Interpolated generalized precision curves based on thorough evaluation measures for all these runs. The curves from the two baselines are clearly on the top but it appears that the baseline with IQE only outperforms the automatic baseline at a recall level lower than 0.2. After this level, the two curves are almost identical. It also appears that runs retrieving XML elements only outperform their full document counterpart at very low recall levels.

As in the INEX 2008 corpus, it appears that IQE based on MWTs can improve document or XML element retrieval whereas AQE does not. This is a surprising difference with our 2008 results [7]. Contrary to the runs we submitted in 2008, this year (2009), we used AQE in all our submitted runs because it had improved performance previously.

Moreover, it appears on these baseline runs that the difference between our automatic baseline run and the one with IQE is not statistically significant. In fact with an Ip[0.01] of 0.56 and a MAiP of 0.28, our baseline run performs much better than in 2008 meanwhile the score of our MWT runs is unchanged. The reason could be the availability this year in the topics of a new topic field with phrases that is used by all of our runs (**ph** feature), including the baselines. This makes the input to the baseline runs somewhat similar to the runs using MWTs. We need to investigate this issue further in order to confirm our intuitions. More experiments are performed hereafter.

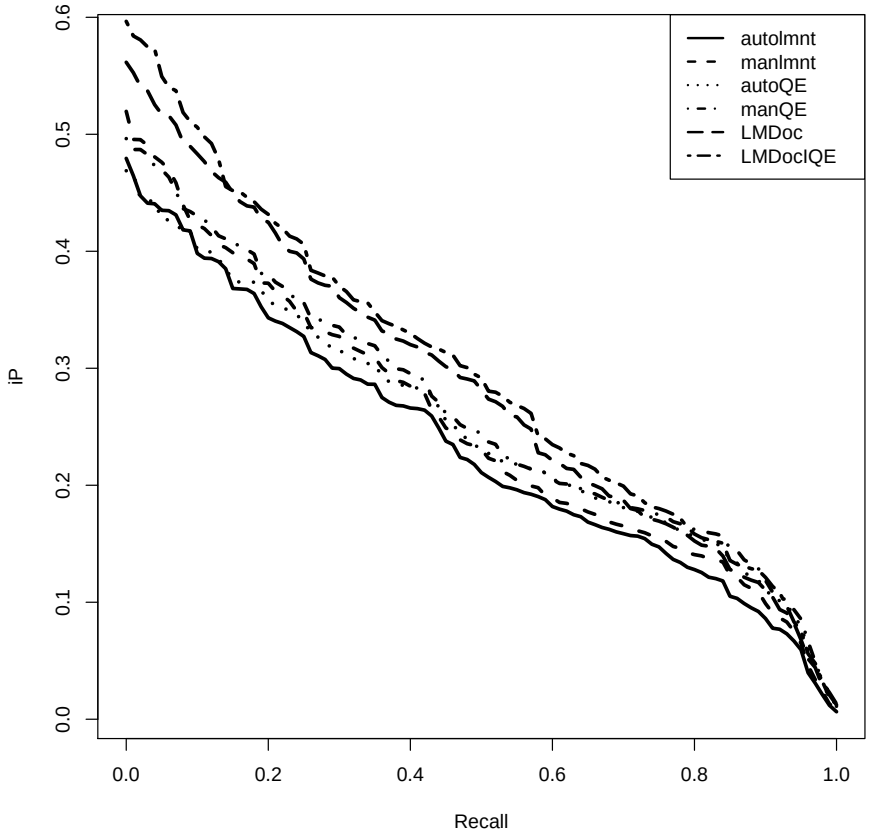

Fig. 1. Interpolated generalized precision curves at INEX 2009 on thorough task.

3.2 The impact of language element models and AQE

First, we explored the impact of two wikipedia XML elements on LM results: document title and paragraphs.

Our idea was to favour documents in which the title is related to the query (shared common terms). We then consider the product of two LM ranking functions, one on the whole document, the other on document titles. It appears that this combination had a clear negative impact on the overall performance as it can be observed on Table 1 and in Figure 1 where the two runs LMDoc and LMDocIQE outperform the two submitted runs autoQE and manQE that only differ on the use of AQE and document titles.

The same observation can be made about AQE. The resulting ranking function is the product of the initial ranking with the expansion. As we already mentioned, on 2008 AQE significantly improved overall performance. In our 2009 runs, it turned out to be a drawback.

For paragraphs we apply the same query on different LM models: one on paragraphs, the other on documents. We then merge the two rankings based on their scores. Since this score is a probability, we wanted to check if it was possible to use it to merge different rankings, despite the fact that these probabilities are estimated on different spaces. This experiment was possible on the thorough task since it allowed overlapping passages.

It appears that considering XML elements does not significantly improve LM baseline. However, on thorough task it appears that iP[0.01] score of full document runs Lyon3LIAmanQE and Lyon3LIAautoQE can be improved by simply merging them with XML element runs as done for runs Lyon3LIAmanlmnt and Lyon3LIAautolmnt.

3.3 Extended baseline runs

As many parameters are combined in our runs, it is necessary to try to isolate the impact of each of them. To do this, we generated more baselines. We then tested successively the effects of stemming and query expansion with MWTs on both the LM and the bag of word models.

Three supplementary baselines based only on the title field and retrieving full documents were generated. No preprocessing was performed on this field. The first baseline **LM** uses language model based on equation 1. The difference between this LM baseline and the previous LMDoc is that we do not use the topic's phrase field. The two other baselines are based on the bag of word models - TFIDF and BM25 using their default parameters in Indri[3] ($k1 = 1.2$, $b = 0.75$, $k3 = 7$). We observed that LM performed better on a non stemmed index whereas TFIDF and BM25 performed better on a stemmed index. Therefore in the remainder of the analysis, we only consider LM runs on non stemmed index and bag of word model runs on stemmed corpus. In Table 2, the "No ph" columns give the scores of the baselines without using phrases from the phrase field whereas "With ph" does the opposite.

	No ph		With ph	
Measure	IP[0.01]	MAiP	IP[0.01]	MAiP
TFIDF	0.6114	0.3211	0.5631	0.3110
BM25	0.5989	0.3094	0.5891	0.3059
LM	0.5389	0.2758	0.5527	0.2826

Table 2. Supplementary baselines based on bag of words model and Language Model.

From these results, it appears that the strongest baseline is TFIDF followed by BM25. What is surprising is that with a MAiP of 0.32, this baseline outperforms all official runs submitted to the thorough task and with an $iP[0.01]$ of 0.61138, it would have been ranked fifth (and third by team) in the focused task.

[3] http://www.lemurproject.org/doxygen/lemur/html/IndriParameters.html

A t-test shows that the difference between TFIDF and BM25 is not statistically significant, but they are between these bag of word models and LM for both IP[0.001] and MAiP (p-value < 0.01 for TFIDF and < 0.05 for BM25).

We now investigate the impact of MWTs to expand the seed query on the performance of the language model (LM) and the bag of word models, without any tuning to account for the nature of MWTs. First, we only consider MWTs from the topic's phrase field. The resulting runs are still considered as automatic because there is no manual intervention. We shall refer to them as **ph** runs. It appears that this incremental process handicaps both TFIDF and BM25 as it can be observed in the columns "with ph" of Table 2. By adding phrases, the scores for TFIDF drop even more than BM25's although the latter's scores also drop. On the other hand, the LM model naturally takes advantage of this new information and each addition of MWTs improves its performance. LM with ph corresponds to our initial automatic baseline LMDoc. It appears that difference between LMDoc and the best TFIDF scores is not more significant after adding MWTs in topic phrase field. Adding more MWTs following our IQE process only generates more noise in TFIDF and BM25 runs, but improves LM performance, even though the improvement is not statistically significant.

4 Discussion

We used Indri with Dirichlet smoothing and we combined two language models, one on the documents and one on elements. The results from both models are then merged together and ranked by decreasing probability.

For query representation, we used NLP tools (summarizer and terminology extraction). We started from the topic phrase and title, then we added related Multiword Terms (MWT) extracted from the other topic fields and from an automatic summary of the top ranked documents by this initial query. We also used standard Automatic Query Expansion when applied to the document model.

Other features tested are the Indri operators to allow insertions of words (up to 4) into the MWTs and favoring documents in which the MWTs appear in the title.

As observed on the previous INEX corpus (2008), IQE based on MWTs still improves retrieval effectiveness but only for search strategies based on the language model. On the contrary, automatic query expansion (AQE) using the same parameters has the reverse effect on the 2009 corpus. At the baseline level, we observe that TFIDF performs significantly better than LM, but LM naturally allows an incremental and interactive process. This suggests that users can more easily interact with an LM in an IR system. For two consecutive years, we have observed that stemming does not have any significant impact on the performance of the different strategies we have tested. Another interesting finding in this year's experiments is that baselines generated using the bag of word models with their default parameters as implemented in Indri and without recourse to any NLP (MWTs) nor to query expansion, outperformed language models that combined interactive query expansion based on MWTs. We need to investigate

these findings further. We also need to ascertain the exact impact of the phrases furnished this year to represent topic's contents with regard to MWTs that we exctracted from other sources.

References

1. Perez-Carballo, J., Strzalkowski, T.: Natural language information retrieval: progress report. Information Processing and Management **36**(1) (2000) 155 – 178
2. Vechtomova, O.: The role of multi-word units in interactive information retrieval. In Losada, D.E., Fernández-Luna, J.M., eds.: ECIR. Volume 3408 of Lecture Notes in Computer Science., Springer (2005) 403–420
3. Knoth, P., Schmidt, M., Smrz, P., Zdrahal, Z.: Towards a framework for comparing automatic term recognition methods. In: ZNALOSTI 2009, Proceedings of the 8th annual conference, Vydavatelstvo STU (2009) 12
4. Jones, K.S., Walker, S., Robertson, S.E.: A probabilistic model of information retrieval: development and comparative experiments. Inf. Process. Manage. **36**(6) (2000) 779–840
5. Kageura, K.: The dynamics of Terminology: A descriptive theory of term formation and terminological growth. John Benjamins, Amsterdam (2002)
6. Ibekwe-SanJuan, F.: Constructing and maintaining knowledge organization tools: a symbolic approach. Journal of Documentation **62** (2006) 229–250
7. Ibekwe, F., SanJuan, E.: "use of multiword terms and query expansion for interactive information retrieval". In Geva, S., Kamps, J., Trotman, A., eds.: INEX 2008 (selected papers), LNCS 5631, Berlin Heidelberg, Springer-Verlag (2008) 54 – 64
8. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, New York, NY, USA, ACM (1998) 275–281
9. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. ACM Trans. Inf. Syst. **22**(2) (2004) 179–214
10. Metzler, D., Croft, W.B.: Combining the language model and inference network approaches to retrieval. Information Processing and Management **40**(5) (2003) 735–750
11. Strohman, T., Metzler, D., Turtle, H., Croft, W.B.: Indri: A language-model based search engine for complex queries (extended version). IR 407, University of Massachusetts (2005)